Neural Network Ensembles for Time Series Prediction

Dymitr Ruta and Bogdan Gabrys

***Abstract*— Rapidly evolving businesses generate massive amounts of time-stamped data sequences and defy a demand for massively multivariate time series analysis. For such data the predictive engine shifts from the historical auto-regression to modelling complex non-linear relationships between multi-dimensional features and the time series outputs. In order to exploit these time-disparate relationships for the improved time series forecasting, the system requires a flexible methodology of combining multiple prediction models applied to multiple versions of the temporal data under significant noise component and variable temporal depth of predictions. In reply to this challenge a composite time series prediction model is proposed which combines the strength of multiple neural network (NN) regressors applied to the temporally varied feature subsets and the postprocessing smoothing of outputs developed to further reduce noise. The key strength of the model is its excellent adaptability and generalisation ability achieved through a highly diversified set of complementary NN models. The model has been evaluated within NISIS Competition 2006 and NN3 Competition 2007 concerning prediction of univariate and multivariate time-series. It showed the best predictive performance among 12 competitive models in the NISIS 2006 and is under evaluation within NN3 2007 Competition.**

I. Introduction

Recent e-revolution has led to the situation in which most of business and organisation entities continuously generate massive amounts of data which de facto constitute multivariate time series. Prediction of such time series is extremely important and vital for surviving and gaining competitive advantage in case of businesses and simply gives the awareness and time to prepare for what is about to be present in general case. Time series prediction is a very challenging signal processing problem as in real situations it is typically a function of a large number of factors most of which are unknown or inaccessible at the time of prediction. Although such time series appear as very noisy, non-stationary and non-linear signals, its history carries a significant evidence that can be used to build the predictive model [1],[2],[3].

A number of techniques have been developed in an attempt to predict time series in various contexts typically in financial trading, energy and water distribution, chemical processes monitoring but also in various sociological and many other problems. Starting from a simple linear Autoregressive Moving Average models (ARMA) [2] through conditional heteroscedastic models like ARCH or GARCH [2] up to the complex non-linear models [2],[4], the idea is similar: establish the regression-based description of the future series based on the historical data series. More recently a number of machine learning techniques started to be applied to time series forecasting and on a number of occasions showed considerable improvement compared to traditional regression models [5], [3], [6]. Neural networks are particularly good at capturing complex non-linear characteristics of time series [5], [3]. Support vector machine represents another powerful regression technique that immediately found applications in time series forecasting [7],[6].

While there is an extensive knowledge available in machine learning and pattern recognition, it has been rarely used in temporal aspects of time series prediction. The major problem for classification models lies in their inability to predict continuous numerical values whereas advanced nonlinear regression models mostly designed for static problems could not properly handle the temporal aspect of time series.

This work presents a composite regression model based on an ensemble of Neural Networks that is designed to comprehensively handle time series in various operational scenarios and generate robust predictions. The model has been evaluated on the course of two international time series forecasting competitions in one of which it generated the best prediction results out of 12 competitive models worldwide.

A. NISIS Competition 2006

The objective of the NISIS 2006 competition was to create an adaptive mathematical model capturing the relationship between 14 input variables and an output variable describing catalytic oxidisation process in the multi-tube reactor. The input variables represent various measures continuously collected during the chemical process like: flow of air/gasses [kg/hr], temperature and various concentration measures, whereas the output variable represents the catalytic activity of the process. All the variables vary over time effectively forming a multivariate time series. There was no restriction concerning model selection but an adaptive component of the predictive model was an obligatory requirement.

In the first phase of the competition 8 months of data (242 * 24 hours), both input $\mathbf{x}$ and output y was given along with next months input data to create the model and make predictions for the catalytic activity over the next month. Then on submission of the predictions the true output values are given for this month along with the input data for the next month and the process repeats over the period of 4 months as shown in Figure 1. The model was evaluated using $N = 15$ output predictions per months representing the activity measured at the end of every second day and the overall error rate is calculated as follows:

Dymitr Ruta, British Telecom Group CTO, Intelligent Systems Lab, Adastral Park, Orion MLB1 PP12, Martlesham Heath IP5 3RE, UK (phone: +44(0)1473216047; fax: +44(0)1473623683; email: dymitr.ruta@bt.com).
Bogdan Gabrys, Bournemouth University, Computational Intelligence Research Group, Poole House, Talbot Campus, Fern Barrow, Poole, Dorset, BH12 5BB, UK (phone: +44(0)1202965298; fax: +44(0)1202965314; email: bgabrys@bournemouth.ac.uk).

1-4244-1380-X/07/$25.00 ©2007 IEEE

$$ERR = \sum_{j=1}^{4} \frac{100}{N} \sum_{i=1}^{N} \frac{|y_{true} - y_{pred}|}{|y_{true}|} \quad (1)$$

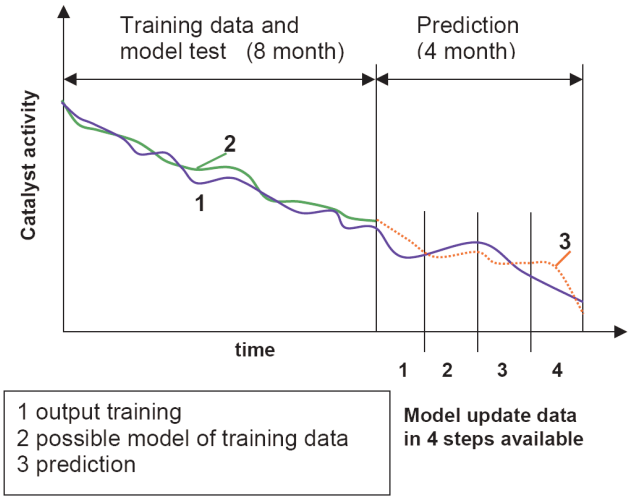

Fig. 1. Visualisation of the NISIS 2006 Competition 2006 task.

B. NN3 Neural Network Forecasting Competition 2007

The objective of this competition was to build a model which would be able to generate predictions for a number of univariate time series up to 18 points into the future from the end of available series. The challenge in this competition lies in using the same model for a number of very different series some of which are very volatile other clearly exhibit periodicity whereas all the series contain a significant noise component. A snapshot of selected time series of this competition is shown in Figure 2. Overall there was 11 time series of about 120 points and for each the future 18 points of the series had to be predicted. The predictions are to be evaluated using symmetric mean absolute percent error defined as follows:

$$ERR = \sum_{j=1}^{11} \frac{|y_{true} - y_{pred}|}{(y_{true} + y_{pred})/2} \cdot 100 \quad (2)$$

II. Feature Generation and Selection

The problem of time series prediction is merely a problem of extracting a manageable set of good features. The temporal dimension of the data multiplies the potential scope of M-feature space effectively stretching it up to $M \cdot L$ dimensions where L stands for the length of time series. What it means is that whatever set of M features describes the actual problem, its temporal variability enforces to consider also the whole available history of feature series as potential inputs to the predictive model. Careful selection of features is therefore of much greater importance compared with the static-data prediction problem. On the other hand temporal feature selection depends strongly on the availability of features in their temporal relation to outputs as well as the depth of outputs prediction. To take full advantage of these characteristics the prediction problem has to be considered within an appropriate temporal prediction paradigm.

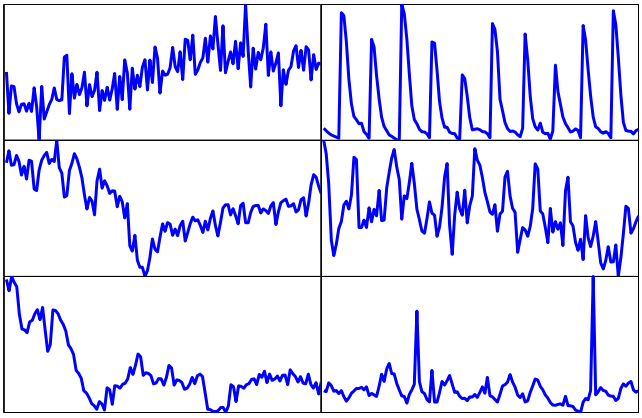

Fig. 2. Selected time series from the NN3 Competition.

A. Temporal Prediction Paradigm

In defining the temporal prediction paradigm the two important questions need to be answered: whether and how deep history of inputs and outputs is to be used for prediction. A good starting point for this consideration is to initially assume that all available evidence can be used for predictions i.e:

$$y_t = f(\mathbf{x}_t, \mathbf{x}_{t-1}, .., \mathbf{x}_{t-L}, y_{t-1}, ..., y_{t-L}) \quad (3)$$

where $\mathbf{x}_t$ is a vector of current inputs, y_t the current output and L denotes the length of the time series. If only $\mathbf{x}_t$ is available the problem reduces to a static regression allowing to only extract the output from current inputs yet without the ability to predict future output series. On the other end if only the output series $y_t, ..y_{t-L}$ is available the problem turns into a standard auto regression trying to predict future series solely from its historical evolution. The most comprehensive predictive model as given by Eq. 3 would use all the evidence available, yet not necessarily result in the best performance. In general a dedicated feature selection process has to be applied to the pool of all available features in order to extract optimal subset of features i.e. the subset for which trained model shows the highest predictive performance.

Another point to consider when selecting features is the depth of predictions. Given that predictions are generated from current or past feature values, if the depth of predictions reaches out multiple periods ahead the further predictions would have to use previous predictions as inputs. The accumulation of noise or other residual inaccuracies could very quickly make subsequent predictions highly inaccurate. The problem complicates even more for multivariate time series with many inputs $\mathbf{x}_t$. Generation of deep output predictions y_{t+d} where d is the prediction depth depends in that case on the prior prediction of inputs $\mathbf{x}_{t+k}$, $1 \leq k \leq d$ if those are used as features for predictive model of output.

B. Feature Selection

The available data for NISIS competition constitutes a timestamp column, 14 continuous variables and the continuous output variable taking values within the range (-1,1). For this particular problem the current inputs are always available and therefore even for deep predictions the predictive model could use true current features. To a certain degree it is a reconstruction or recognition of the output from always available inputs. Whether to include or not the histories of inputs and output into the set of features the model will be trained on is yet to be determined.

For NN3 competition the choice of features was much narrower. For each series the only option available was to select features built from the history of the same output series. In that case, however, any further-looking predictions had to use its own predictions from previous steps prompting problems of accumulated noise mentioned in Section I.

For both problems we applied the same rather simplistic feature selection strategy based on semi-exhaustive search. The first method generated features from equally spaced historical subsequence defined by the following equation:

$$f_i = \mathbf{x}_{t-i \cdot step}, i \in [1, 2, ..., N] \quad (4)$$

The maximum number of features was set as a parameter and fixed at $N = 100$ for both problems to avoid lengthy model training processes.

The second selection method assumed greedy extension of the feature set by subsequently adding a single feature $f_i = x_{t-i}$ provided it reduces the error rate of the predictive model built on the appended feature set. If not, then the feature set is not extended and the f_i feature is replaced with the following f_{i+1} feature that undergoes the same performance testing process. The selection terminates when history depth index i reaches the upper limit arbitrarily set as in previous selection method to $N = 100$. If x_t is a vector: $\mathbf{x}_t$ then the process of searching extends at each time step by checking subsequently all individual j component of a vector x^j_{t-i}.

All the configurations of steps and feature set size $n \leq N$, for the first method and all incrementally appended subsets for the greedy selection method have been evaluated exhaustively within the limits of data availability using the model prediction rate criterion. Surprisingly for the NISIS competition data all the configurations of historical features for both selection methods were worse than the actual current set of inputs $\mathbf{x}_t$. Even addition of past outputs did not improve the performance obtained for current inputs $\mathbf{x}_{t+i}$. In case of the univariate series from NN3 competition, the two selection methods led to very different solutions across different timeseries with the greedy method clearly resulting in a better predictive model performance. Details of the selected feature subsets and the resulting predictive model performances are shown in the experimental Section IV.

III. Neural Networks Ensemble

Due to continuous nature of the output variables in both problems the choice of predictive model narrows down to the multiple-input one-output temporal regression problem. Neural Networks are considered to be a universal non-linear regression model with the ability to control its complexity and high predictive diversity that can be further encouraged by varying network architecture and initialisation conditions, cross-training and even simple injection of noise to the data [8]. Technically, an individual neural network represented a Feedforward Multilayer Perceptron structured within three hidden layers sized up to : [32 32 32] which is trained using an efficient iRPROP+ algorithm [11] that scales linearly with the number of parameters to be optimised.

Given many diverse and well performing predictors it is possible to construct an ensemble of regressors that would jointly outperform any individual regression model. There is many ways the individual NN models can be combined in the ensemble: the simplest is just by averaging the individual network outputs, the other method often used is a linear combination of NN outputs [9], [10]. Although complexity of such model dramatically increases, given the performance critical nature of the predictive task and relatively small data set such model becomes a viable and analytically strong proposition.

A. Model Diversification

To encourage better generalisation ability of the overall model a number of diversification strategies have been applied to increase the complementarity of the constituent ensemble member models. Diversification was applied at all stages of the individual NN model building process from varying its internal architecture and initialisation condition, through training on different noise-injected data subsets up to varying the number of epochs after which the iRPRP+ learning [11] terminates the model building process.

The selected number M of such NN models is first assigned with different randomly selected architectures of the hidden layers not exceeding in size the limits set at each layer by [32 32 32] and randomly initialized weights. Then all the models are cross-trained and evaluated on many different partitions of the available training data following the k-fold cross-validation scheme and using slightly different number of epochs after which the model learning process terminates. On average M/k models are trained on data from a single k-fold cross-validation partition and are assigned the error rate obtained on the testing part of this partition. Overall the training process involves M individual NN learning processes and scales linearly with the size of ensemble.

B. Model selection

Following the training and evaluation process a two-stage model selection is applied to construct the final ensemble. In the first stage the fixed fraction of the best models from each split are selected according to their regression testing error. Then the models selected in the first stage are pooled together and again the selection guided by the individual error rate proceeds to give the final ensemble with the desired number of NN models for simplicity limited to 6 in our experiments.

C. Predictions Smoothing and Adaptability

Early experiments with the ensemble of neural networks indicated that the predicted series on average follows quite well the true series, although it exhibit a significant noise component. To reduce the impact of noise an original smoothing technique has been applied to the signal composed of the predicted series obtained for the training series and directly following validation series that has not been used during the ensemble training process and which was formed of about a quarter of left out training set. Such predicted signal is compared with the corresponding true output series to build an optimised smoothing of the predicted time series. The smoothing model applied to the data comes in two stages. First a procedure called $\mathrm{signal\_filter(x,k,r)}$ is used to remove high-level noise component. This procedure compares the predicted signal with the bi-directional k-step moving average of this signal and replaces the original signal with the aggregated signal where the difference between the 2 signals is greater than r times standard deviation of the original signal. The resulting signal is further smoothed using the same bi-directional n-step moving average yet in generally using different step parameter of the aggregation.

K-Step bi-directional smoothing is a simple procedure applicable to time series $y(t)$ where $t = 1..,N$, which returns smoothed series $y'(t)$ such that:

$$\begin{cases} y'_t = \frac{1}{2k-1}\sum_{i=t-k+1}^{t+k-1} y_i & \text{for} \quad t = k,..,N-k+1 \\ y'_t = \frac{1}{t+k-1}\sum_{i=1}^{t+k-1} y_i & \text{for} \quad t = 1,..,k-1 \\ y'_t = \frac{1}{N-t}\sum_{i=t-k+1}^{N} y_i & \text{for} \quad t = N-k+2,..,N \end{cases} \tag{5}$$

The three smoothing parameters: k,r and n are optimised with respect to the regression error rate obtained for the validation set via a naive looping through all possible combinations within the grid of 20 values per parameter giving the total of 8000 evaluations. The trained ensemble of NN models along with optimised smoothing parameters constitute the fully trained model ready to make predictions.

To provide certain level of model adaptability the training process was fixed to the the data dynamically changing according to the fixed-width moving window scheme. Assuming that the available training series finishes at time t the model was trained on the series of the length w ranging from $t-w+1$ to t. The window width w has been optimised via a naive evaluation of the whole range of grided window widths within the reasonable limits specific to each series. The conceptual scheme capturing all the step of the model building process is depicted in Figure 3.

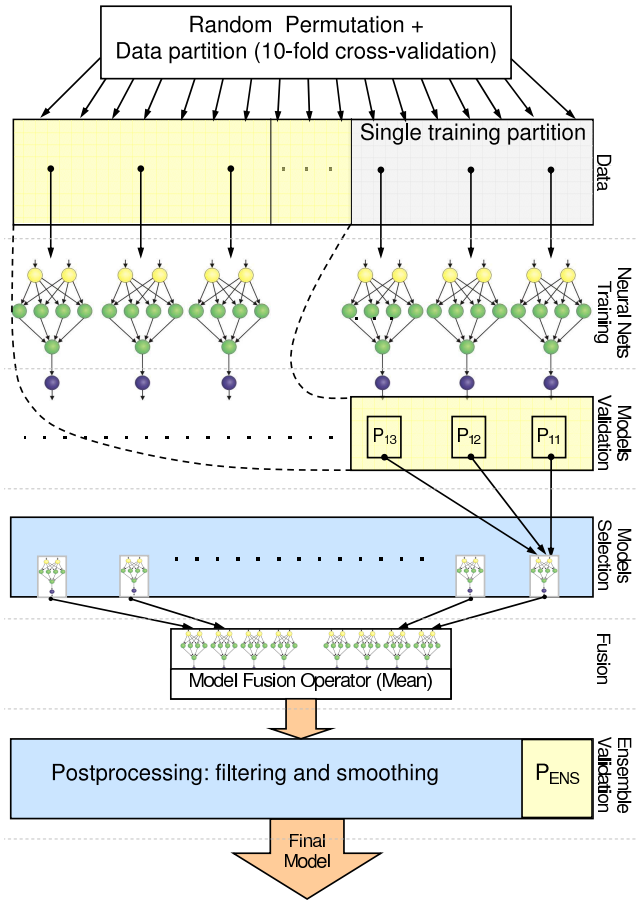

Fig. 3. NN ensemble model building process

IV. Experiments

The experimental part of this work forms the submission of the model predictions to the NISIS Competition 2006 and in parts experiments carried out on the training part of the NN3 Neural Network Forecasting Competition 2007. As discussed in Section I the prediction of the catalyst activity time series for NISIS 2006 was organised in 4 monthly slots of the following 8 months of complete input and output training data.

The optimisation of the moving window width for adaptive learning showed that 8 months is the optimal window width, which suggests that the available data is still not in excess for training and at this stage it appears that the more data used for training the better the performance. Using this 8-months window the presented model was subsequently rebuilt on the 8 months of preceding data using first 6 months for proper training and the remaining two months for validation and fine-tuning of the smoothing parameters as shown in the example in Figure 4. Interestingly, at each iteration of the model testing process, the model found significantly different smoothing parameters. Initially the optimal set of smoothing parameters was $(11, 0.9, 56)$, yet for the last prediction phase the optimal parameters turned out to be $7, 0.8, 231)$. It proves that whereas the global analytical engine of the model inherited from a mixture of neural networks remains data greedy, the smoothing parameters become very sensitive to the local variability of the data and hence accommodate the adaptive component of the model tuning. The complete numerical performance comparison of all the models participating in the NISIS Competition 2006 is shown in Table I. Note that the presented model outperformed immensely other competitors leaving the second best model with twice as large an error rate. It is important to note that these prediction results have been obtained using current input series as features

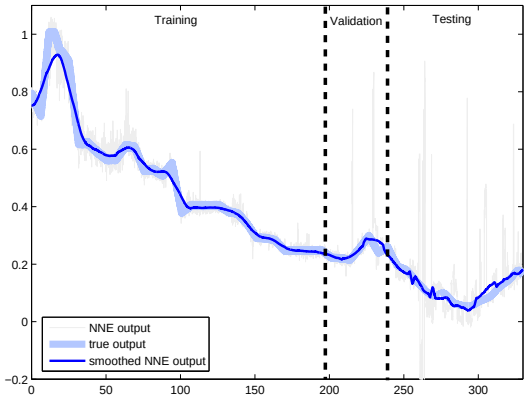

Fig. 4. Model building process for NISIS Competition 2006

only. As mentioned in Section II-B all combinations using historical inputs and outputs resulted in a significantly worse performance which due to a lack of space is not shown here.

TABLE I
ERROR RATES [%] COMPARISON FOR ALL MODELS SUBMITTED TO NISIS 2006 COMPETITION

Rank	Month 1	Month 2	Month 3	Month 4	Average
1	**20.13**	**21.01**	**12.87**	**19.14**	**18.28**
2	43.41	18.38	52.31	24.14	34.56
3	63.15	17.83	33.89	28.91	35.94
4	62.56	14.48	53.75	37.05	41.96
5	81.46	29.80	33.78	23.06	42.02
6	59.01	69.91	37,21	27.54	48.41
7	24.13	87.15	54.16	28.74	48.54
8	75.97	25.48	80,20	27.77	52.35
9	77.78	84.80	32.24	22.21	54.25
10	47.87	126.71	32.59	39.45	61.65
11	52.56	135.91	35.46	57.95	70.47

For the NN3 2007 competition apparently simpler problem required in fact more complex classifier selection process. For all 11 time series of about 120 data points each exactly the same model building process has been applied to generate 11 fine-tuned ensembles with a set of optimised smoothing parameters each. The predictions obtained for the training and validation sets are shown in Figure 5 in 2 versions corresponding to the feature selection method used. The first column represents predictions obtained using equidistant stepped subsequence series whereas the second column shows the predictions obtained using greedy incremental additions of historical series as described in Section III-C. The predicted signal is overlayed on the thick lighter true signal for comparison. The numerical mean error rate for only the validation set is also shown in Table III. For most of the series the predicted signal is very close to its original values, which is the case for both the training part and validation part which was not used to train the ensemble of neural networks. The model sometimes struggles with sudden sharp signal peaks up to multiple standard deviations over its mean. The problem with such peak is that it is not clear whether the peak is a genuine signal behaviour that could influence the future or it is just a sharp noise impulse. Driven by the overall error minimisation the model tends to ignore the infrequent out-of-pattern sharp impulses, yet sometimes tend to successfully accommodate repetitive peaks even if the signal rises above 3 standard deviations above the mean in a single time step as shown for the 3^{rd} time series (TS-3). The results also demonstrated that the greedy incremental method for feature selection tends to produce better results.

Finally, looking into the selected features and the optimal smoothing parameters it has to be said that the models were very different for each time series. Some statistics describing the selected feature sets along with the smoothing parameters are shown in Table II.

V. CONCLUSIONS

This work promotes a new model for time series prediction which combines highly robust ensemble of neural network regressors with the intelligent smoothing of highly noisy output signal. The individual MLP type neural networks are diversified by forcing different internal architecture and weight initialisation models and are cross-trained on different partitions of the training data with injected noise component to further boost the generalisation abilities of the final ensemble. The model building process is supported by the simple yet effective greedy feature generation method and the predicted output signal is further validated using original smoothing technique to remove excessive noise component. The model has been tested on the course of 2 International competitions for time series predictions. It has won NISIS 2006 Competition leaving the second-best model with twice as large an error rate and is under evaluation for the NN3 2007 Competition for which it shows robust prediction results across many very different time series.

REFERENCES

[1] T. Dietterich and R. Michalski. Learning to predict sequences. In Machine Learning: An Artificial Intelligence Approach, vol. 2, Morgan Kaufmann, 1986
[2] R.S. Tsai, *Analysis of financial time series*, John Wiley & Sons, 2002.
[3] C.L. Giles, S. Lawrence and A.C. Tsoi, *Noisy time series prediction using recurrent neural networks and grammatical inference*, Machine Learning vol. 44(1/2), pp. 161–183, 2001.
[4] M. Casdagli, *Nonlinear prediction of chaotic time series*, Physica vol. 35, pp. 335–356, 1989.
[5] Z. Vojinovic, V. Kecman, R. Seidel, *A data mining approach to financial time series modelling and forecasting*, International Journal of Intelligent Systems in Accounting, Finance & Management vol. 10(4), pp. 225–239, 2001.
[6] F.E.H. Tay and L.J. Cao, *Modified support vector machines in financial time series forecasting*, Neurocomputing vol. 48(1), pp. 847–861, 2002.
[7] S. Mukherjee, E. Osuna and F. Girosi, *Nonlinear prediction of chaotic time series using support vector machines*, In Proc. of the 7^{th} IEEE Workshop on Neural Networks for Signal Processing, Amelia Island, FL, IEEE Press, pp. 511–520, 1997.
[8] A.J.C Sharkey, *Combining Artificial Neural Nets: Ensemble and Modular Multi-Net Systems*. Springer-Verlag, London, UK, 1999.
[9] A.J.C Sharkey and N.E. Sharkey, *Combining diverse neural nets*, The Knowledge Engineering Review vol. 12(3), pp. 231–247, 1997.
[10] S. Hansen and P. Salamon, *Neural network ensembles*, IEEE Trans. on Patt. Analysis and Machine Intelligence 12(10), pp. 993–1001, 1990.
[11] C. Igel and M. Husken, *Improving the Rprop Learning Algorithm*, Proc. of the 2nd Int. ICSC Symposium on Neural Computation, pp. 115–121, 2000.

TABLE II

OPTIMISED MODEL PARAMETERS INCLUDING FEATURE SET STATISTICS AND SMOOTHING PARAMETERS FOR 11 TIME SERIES.

Time series	TS-1	TS-2	TS-3	TS-4	TS-5	TS-6	TS-7	TS-8	TS-9	TS-10	TS-11	Mean
Feature set size	7	7	5	7	10	4	12	4	10	7	4	7
Mean feature lag	6.28	16.14	19	9.28	14.3	3	13.91	6.25	13.6	12	11.5	11.58
Min feature lag	1	1	1	1	1	1	1	1	1	1	1	1
Max feature lag	18	49	36	19	40	6	32	13	29	41	40	29.36
Smoothing par. k	4	1	1	1	7	14	9	2	15	7	2	5.72
Smoothing par. r	0.44	0.01	0.01	0.01	0.01	0.01	0.01	0.48	0.01	0.17	0.11	0.11
Smoothing par. n	1	1	1	1	1	2	1	2	3	6	17	3.27

TABLE III

ERROR RATES [%] COMPARISON FOR ALL 11 TIME SERIES WITHIN NN3 2007 COMPETITION OBTAINED USING NEURAL NETWORK ENSEMBLE WITH 2 DIFFERENT FEATURE SELECTION METHOD.

Selection Method	TS-1	TS-2	TS-3	TS-4	TS-5	TS-6	TS-7	TS-8	TS-9	TS-10	TS-11	Mean
Equidistant Step	2.16	12.17	49.73	14.71	2.30	4.65	2.98	15.93	3.37	67.73	10.25	16.91
Greedy incremental	2.12	7.17	27.38	7.86	1.21	4.94	2.86	13.73	2.72	69.08	11.78	13.71

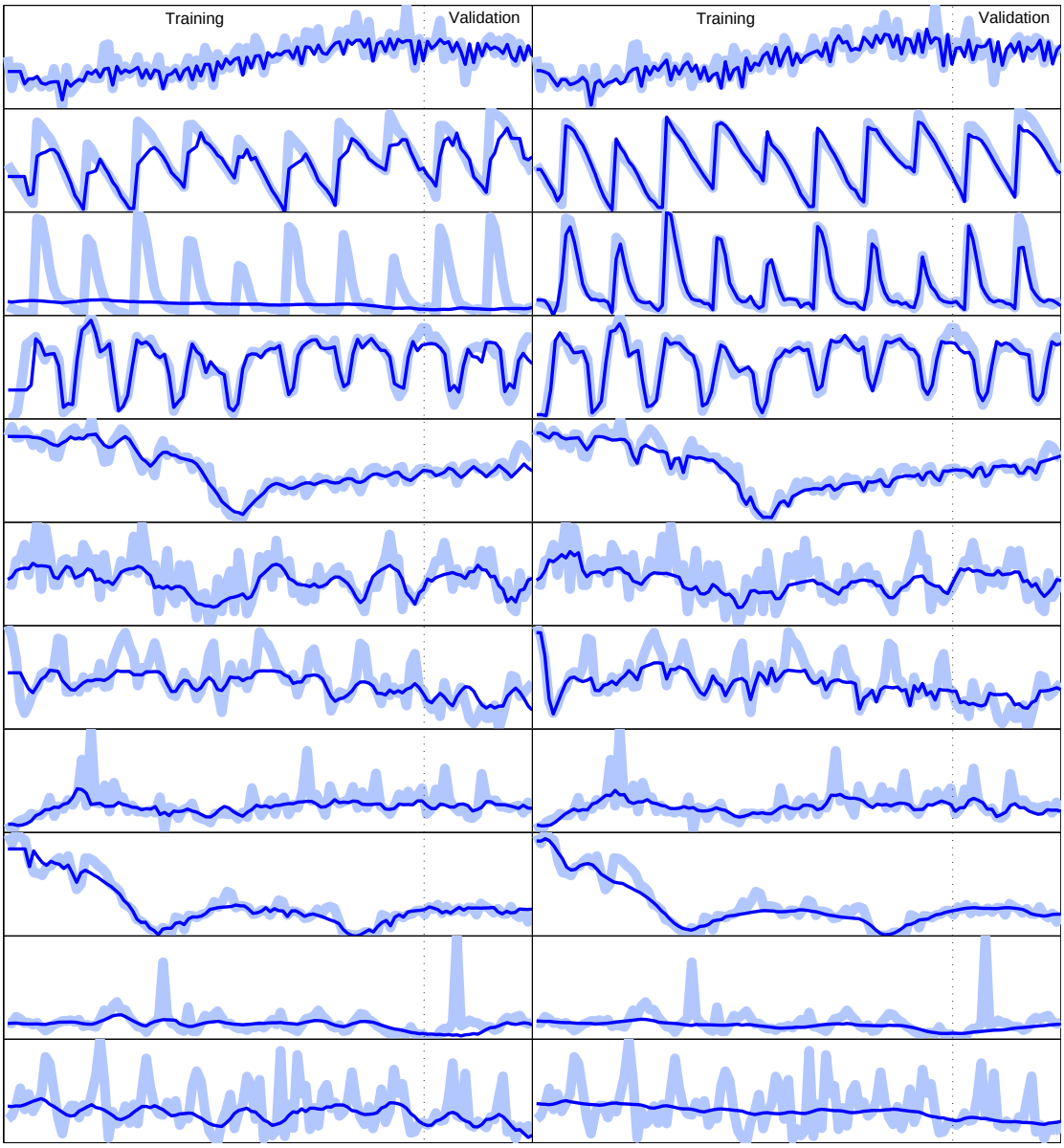

Fig. 5. Comparison of the predictions generated by the NN ensemble for 11 time series of the NN3 Competition for 2 selection strategies: (1^{st} column) equidistant stepped method, (2^{nd} column) greedy incremental method